



TITLE:

行列の条件数の推定(数値計算のアルゴリズムの研究)

AUTHOR(S):

名取, 亮; 塚本, 敦子

CITATION:

名取, 亮 ...[et al]. 行列の条件数の推定(数値計算のアルゴリズムの研究). 数理解析研究所講究録 1983, 483: 212-222

ISSUE DATE:

1983-03

URL:

<http://hdl.handle.net/2433/103420>

RIGHT:

行列の条件数の推定

筑波大電子・情報 名取 亮 (Makoto Natori)

日立 機械研 塚本敦子 (Atsuko Tsukamoto)

1. はじめに

連立1次方程式 $Ax = b$ をガウス消去法で解く際に、行列 A の条件数がわかっていれば、解の精度が推定できる。真の解を x ， β 進 t 桁の演算で得られた値を x_* とすると、

$$\frac{\|x - x_*\|}{\|x_*\|} \leq \text{cond}(A) \beta^{1-t} \quad (1)$$

が成り立つ。行列の条件数は

$$\text{cond}(A) = \|A\| \|A^{-1}\| \quad (2)$$

で定義される。条件数の値はノルムによって異なるが、ここでは L_1 ノルムを考えることにすると、

$$\|A\|_1 = \max_{x \neq 0} \frac{\|Ax\|_1}{\|x\|_1} = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}| \quad (3)$$

であるから、 $\|A\|_1$ は簡単に計算できる。問題は、 $\|A^{-1}\|_1$ をいかに少ない手間で推定するかである。

2. DECOMP の方法

文献[1]の第3章にあるプログラム DECOMP では,

$$\|A^{-1}\|_1 \approx \frac{\|z\|_1}{\|y\|_1} \quad (4)$$

によって推定している。ただし、 y と z は

$$A^T y = e \quad (5)$$

$$A z = y \quad (6)$$

を解いて得られるベクトルで、 e は成分が ± 1 で (5) を解く過程で y の成分ができるだけ大きくなるように符号を選んだものである。

この方法は、つぎのようにして説明できる。行列 A の特異値分解を

$$A = U \Sigma V^T \quad (7)$$

とする。ここで、 U と V は直交行列、 Σ は対角行列で、その対角成分 (特異値) を σ_i とし、

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n$$

であるとする。行列の L_2 ノルムは特異値によってつぎのように表わされる。

$$\begin{aligned} \|A\|_2 &= \sigma_1 \\ \|A^{-1}\|_2 &= 1/\sigma_n \end{aligned} \quad (8)$$

また, 行列 V のオニ列 (特異ベクトル) を v_i とすると

$$A^T A v_i = \sigma_i^2 v_i \quad (9)$$

となる. 一方 (5) と (6) から

$$A^T A z = e \quad (10)$$

だから, z は初期ベクトル e に対して逆反復法を 1 回適用して得られたベクトルであることがわかる. したがって, z は $A^T A$ の最小固有値 σ_n^2 に対する固有ベクトル v_n の成分を多く含むはずである. そこで,

$$z \approx v_n$$

とすると

$$A^T A z \approx \sigma_n^2 z \quad (11)$$

となり, この両辺に左から z^T をかけることにより

$$\|A z\|_2^2 \approx \sigma_n^2 \|z\|_2^2 \quad (12)$$

が得られる. これより

$$\frac{\|z\|_2}{\|y\|_2} = \frac{\|z\|_2}{\|A z\|_2} \approx \frac{1}{\sigma_n} = \|A^{-1}\|_2 \quad (13)$$

となる. このように, DECOMP の方法は本来 L_2 ノルムに対する近似であるが, この関係を L_1 ノルムにおきかえたものが (4) 式で示した DECOMP の推定法である.

ついでに、DECOMPのプログラム中の誤りを指摘しておく。

```

DO 60 KB=1, NM1
-----
DO 55 I=KP1, N
    T = T + A(I, K) * WORK(K)      (*)
55  CONTINUE
    WORK(K) = T                    (**)
-----
60  CONTINUE

```

の(*)と(**)の行は、それぞれ

$$(*) \quad T = T + A(I, K) * \text{WORK}(\underline{I})$$

$$(**) \quad \text{WORK}(K) = T + \underline{\text{WORK}(K)}$$

とするのが正しい。

3. 我々の方法(1)

これに対して、我々は

$$\|A^{-1}\|_1 \approx \|y\|_\infty \quad (14)$$

による推定法を提案する。 y は(5)式により得られるベクトルである。すなわち、

$$A^T y = e$$

である。この方法では z を計算する必要がないので、DECOMPの方法にくらべて計算量を $1/2$ に減らすことができる。この方法は、つぎのようにして説明できる。

一般に、 L_1 ノルムと L_∞ ノルムとの関係は

$$\|A\|_1 = \|A^T\|_\infty$$

であるから

$$\|A^{-1}\|_1 = \|A^{-T}\|_\infty = \max_{x \neq 0} \frac{\|A^{-T}x\|_\infty}{\|x\|_\infty} \quad (15)$$

となる。ここで、 e は $y (= A^{-T}e)$ の成分が大きくなるように選んだベクトルであるから、 $\|A^{-T}x\|_\infty / \|x\|_\infty$ は $x=e$ のとき最大に近くなることが期待される。したがって、

$$\|A^{-1}\|_1 \approx \|A^{-T}e\|_\infty / \|e\|_\infty = \|y\|_\infty \quad (16)$$

が得られる。

実際、 L_∞ ノルムの定義から

$$\|A\|_\infty = \max_{x \neq 0} \frac{\|Ax\|_\infty}{\|x\|_\infty} = \max_i \sum_{j=1}^n |a_{ij}| \quad (17)$$

となる。もし、 $i=m$ のとき $\sum_{j=1}^n |a_{mj}|$ が最大になったとすると、その最大値を与えるベクトル $x=(x_j)$ は、

$$x_j = \text{sgn}(a_{mj})$$

すなわち、成分は ± 1 で符号が A の m 行目の成分の符号に一致するものであることが知られている。 $\|A^{-T}\|_\infty$ の場合は

これを当てはめると、 e の符号が A^{-T} の m 行目すなわち A^{-1} の m 列目の成分の符号に一致するように選ばれていたら、厳密な値を与えることになる。

ここで注目すべきことは、 A が M 行列($A^{-1} > 0$)の場合には、 e の成分をすべて $+1$ にすれば、この方法によって厳密な条件数が求められることである。

4. e の選び方

前に述べたように、 e は成分が ± 1 で、(5)式を解いて y を求める際に y の成分ができるだけ大きくなるように符号を選んだものである。ここでは、符号の選び方をもう少し具体的に述べる。

A をピボットの部分選択を行うガウス消去法によって LU 分解すると、

$$A = (PL)U \quad (18)$$

となる。 L は対角要素が全て1の下三角行列、 U は上三角行列、 P はピボット選択に対応する置換行列である。

$$A^T = U^T (PL)^T$$

であるから、(5)の方程式は

$$U^T x = e \quad (19)$$

$$(PL)^T y = x \quad (20)$$

の2段に分けて解くことができる。(19)は前進代入によって解くことができる。すなわち、 $k=1, 2, \dots, n$ について

$$u_{kk} x_k = e_k - (u_{1k} x_1 + \dots + u_{k-1,k} x_{k-1}) \quad (21)$$

を計算すればよい。ここで、 e_k は ± 1 であるが、符号は上式の右辺の絶対値が大きくなるように選ぶ。そのためには、右辺のカッコでくくった項と異符号になるようにすればよい。この方法は、局所的に見て x_k の絶対値が大きくなる方を選ぶやり方であるが、もう少し先を見越した方法もある(文献[2]参照)。しかし、数値実験によれば、我々の方法では、これによって改善されることは稀である。

5. 我々の方法(2) — 簡便法 —

さらに計算量が少ない推定法として

$$\|A^{-1}\|_1 \approx \|x\|_\infty \quad (22)$$

が考えられる。 x は(19)によって計算されるベクトルである。この方法では、(20)式の計算をしないので、計算量は方法(1)の $1/2$ すなわちDECOMPの方法の $1/4$ で済む。オウ節の議論によれば、

$$\|U^{-1}\|_1 \approx \|x\|_\infty \quad (23)$$

となるので、この方法では $\|A^{-1}\|_1$ を $\|U^{-1}\|_1$ で近似していることになる。

簡単のために (18) 式の PL を L と書くことにすると,

$$A^{-1} = U^{-1} L^{-1}$$

となる。これから

$$\frac{1}{\|L^{-1}\|_1} \leq \frac{\|U^{-1}\|_1}{\|A^{-1}\|_1} \leq \|L\|_1 \quad (24)$$

が得られる。 L の成分を l_{ij} とすると, ピボット選択をしているので, $|l_{ij}| \leq 1$ であるから, $\|L\|_1 \leq n$ である。また, 大抵の場合 L は良条件だから $\|L^{-1}\|_1$ はそれ程大きくはならない。したがって, (24) から $\|A^{-1}\|_1$ と $\|U^{-1}\|_1$ は同程度とみなすことができる。

多少精度を犠牲にしてよければ, この方法は簡便法として有効である。

6. 数値例

いくつかの行列について, DECOMP の方法と我々の方法 (1), (2) による条件数の推定値を計算した。

以下に示す値は, (推定値) / (真値) である。計算は HITAC M-170 (単精度) で行った。

例 1 Frank 行列

$$a_{ij} = n + 1 - \max(i, j)$$

逆行列は三重対角行列で, $\text{cond}(A) = 2n(n+1)$ である。

n	DECOMP	Ours(1)	Ours(2)	条件数(真値)
3	0.81	1.0	0.75	24
4	0.87	1.0	0.75	40
5	0.90	1.0	0.75	60
6	0.92	1.0	0.75	84
7	0.93	1.0	0.75	112
8	0.94	1.0	0.75	144
9	0.95	1.0	0.75	180
10	0.95	1.0	0.75	220

我々の方法(1)は, すべて真の値を与える.

例2 Hilbert 行列

$$a_{ij} = 1 / (i + j - 1)$$

n	DECOMP	Ours(1)	Ours(2)	条件数(真値)
3	0.91	1.0	1.0	7.48×10^2
4	0.76	1.0	0.65	2.84×10^4
5	0.73	0.99	0.99	9.44×10^5
6	0.68	0.87	0.87	2.91×10^7
7	0.09	0.12	0.11	9.85×10^8

すべての場合に, 我々の方法の方が良い値を与える.

$n \geq 7$ では, 条件数が大きすぎるため LU 分解の精度が悪くなり推定値も不正確となる.

例3 文献[3]から選んだ17種の行列

No.	n	DECOMP	Ours(1)	Ours(2)	条件数(真値)
1	3	0.70	0.84	0.77	2.8×10^0
2	4	0.71	1.00	1.00	4.2×10^0
3	6	0.72	1.00	1.00	6.0×10^0
4	4	0.74	1.00	1.00	1.0×10^1
5	4	0.83	1.00	1.00	1.1×10^1
6	3	0.88	1.00	1.00	1.3×10^1
7	4	1.00	1.00	1.00	1.5×10^1
8	4	0.64	0.81	1.04	1.6×10^1
9	4	0.72	1.00	1.00	1.7×10^1
10	4	0.82	1.00	0.40	4.0×10^1
11	3	0.79	0.60	0.60	5.0×10^1
12	3	0.99	1.00	1.00	3.0×10^2
13	4	0.76	1.00	1.00	1.9×10^3
14	4	0.72	1.00	1.00	4.5×10^3
15	3	0.79	1.00	1.00	9.7×10^3
16	2	1.00	1.00	1.00	4.0×10^4
17	2	1.01	1.08	1.08	2.7×10^6

—例 (No.11) を除いて、全て我々の方法の方が正確である。

しかも、ほとんどの場合厳密な値が得られている。

参考文献

- [1] G. E. Forsythe, M. A. Malcolm and C. B. Moler : Computer Methods for Mathematical Computations, Prentice-Hall, 1977
(森正武訳: 計算機のための数値計算法, 科学技術出版社, 1978)
- [2] A. K. Cline, C. B. Moler, G. W. Stewart and J. H. Wilkinson :
An Estimate for the Condition Number of a Matrix,
SIAM J. Numer. Anal. 16 (1979) 368-375
- [3] R. T. Gregory and D. L. Karney : A Collection of
Matrices for Testing Computational Algorithms,
Wiley & Sons, 1969